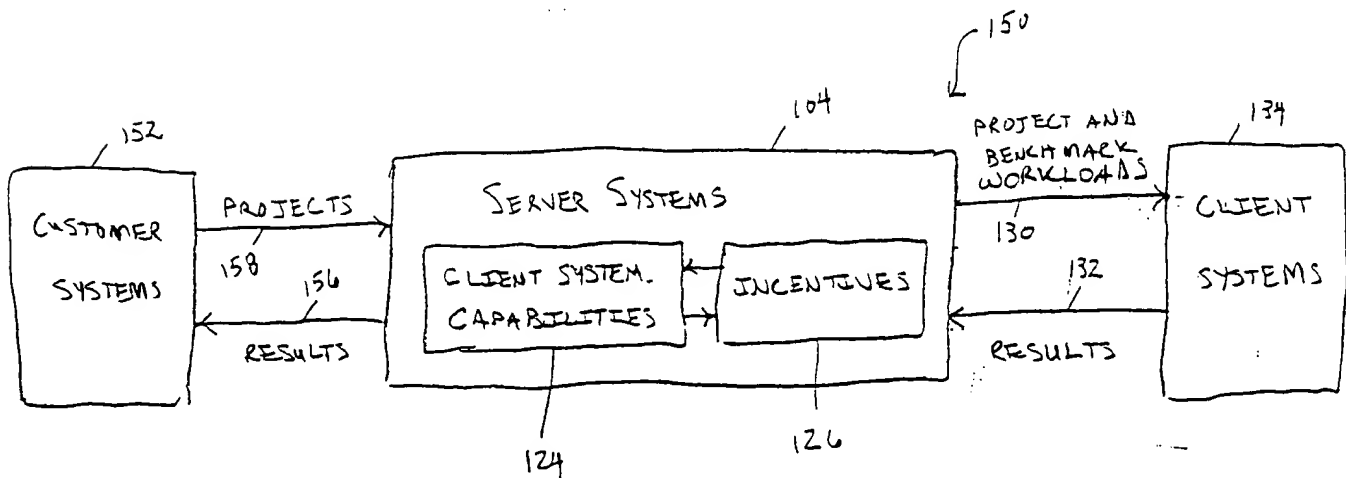


FIG. 1A



00502803-062300

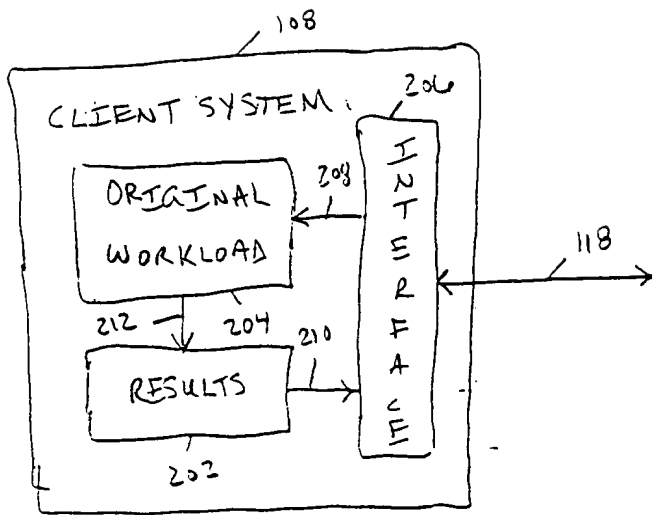


FIG. 2A

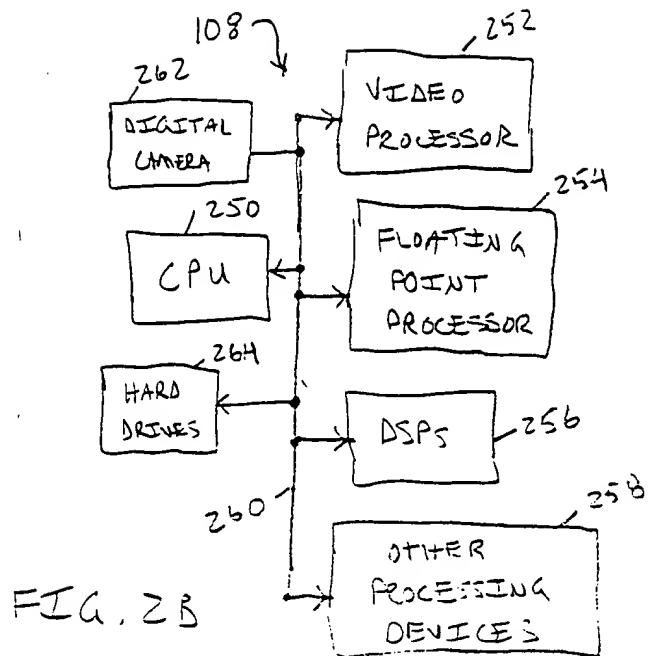


FIG. 2B

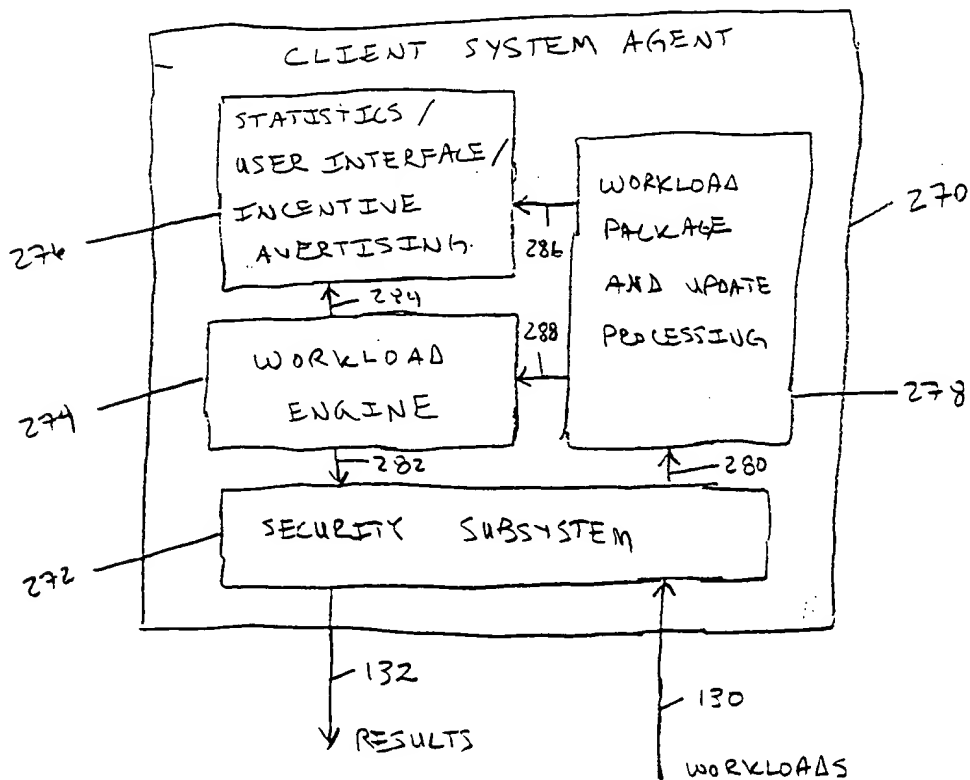


FIG. 2C



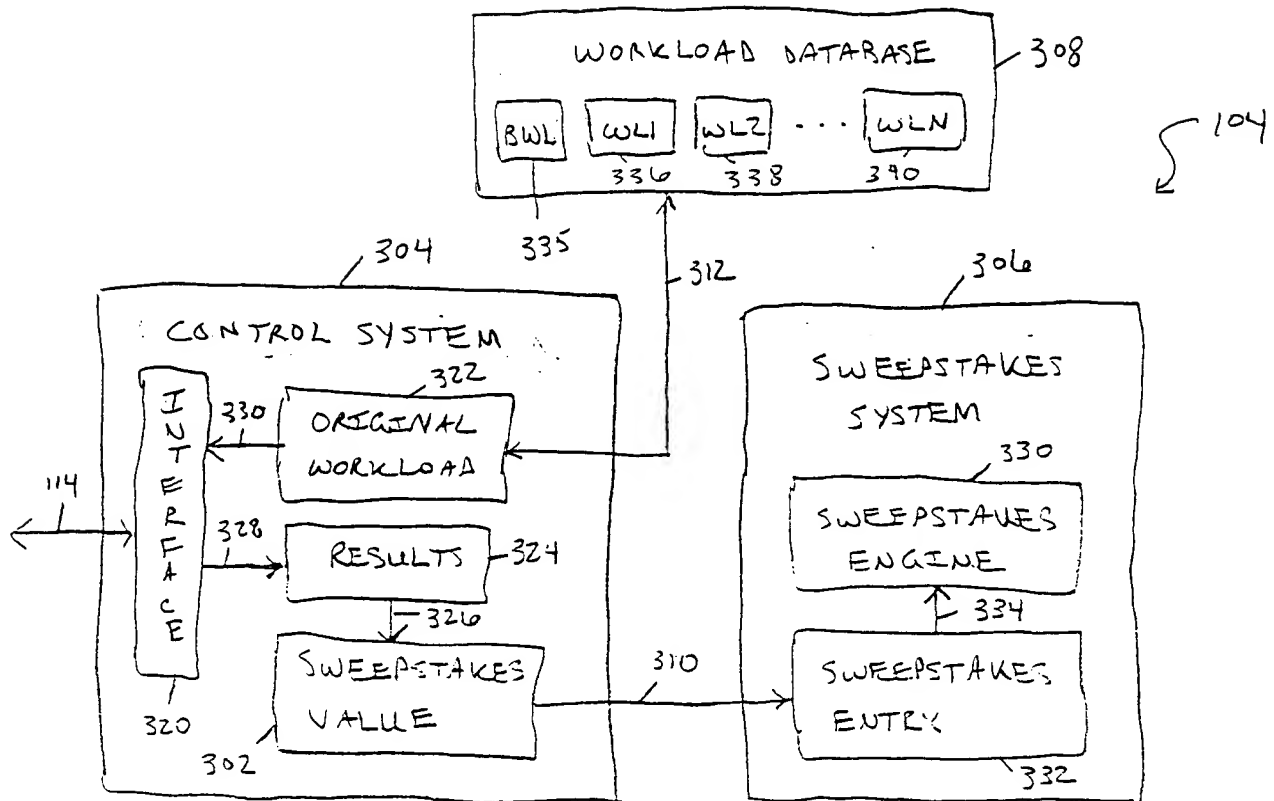


FIG. 3A

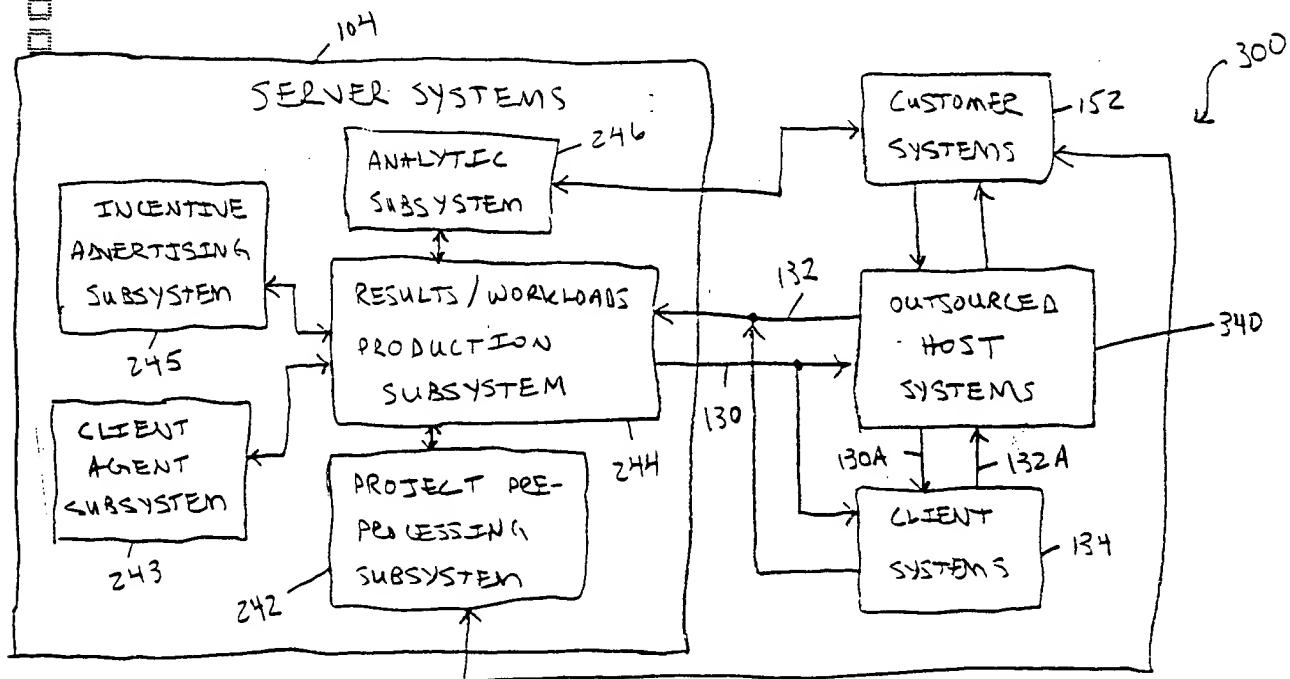


FIG. 3B

05602603-062300

The diagram illustrates a workload management system architecture with the following components and data flow:

- WORKLOAD MANAGER (358)**: The top-level component that receives **350** (Workload) and manages the system.
- APIs (356)** and **DATA PARSER (352)**: These components interact bidirectionally with the Workload Manager and the Security Subsystem.
- SECURITY SUBSYSTEM (354)**: A central component that interacts with the APIs, Data Parser, and the Agent Abstraction Layer.
- AGENT ABSTRACTION LAYER (360)**: The bottom component that receives **132** (Results) and outputs **130** (Workloads).

The flow of data is as follows: **350** (Workload) enters the **WORKLOAD MANAGER (358)**. The **WORKLOAD MANAGER (358)** interacts with **APIs (356)** and **DATA PARSER (352)**. Both **APIs (356)** and **DATA PARSER (352)** interact with the **SECURITY SUBSYSTEM (354)**. The **SECURITY SUBSYSTEM (354)** interacts with the **AGENT ABSTRACTION LAYER (360)**. The **AGENT ABSTRACTION LAYER (360)** receives **132** (Results) and outputs **130** (Workloads).

The diagram illustrates the Security Subsystem architecture. It consists of three main components: a DATA PARSER (352), a WORKLOAD MANAGER (358), and a SECURITY SUBSYSTEM (354). The DATA PARSER and WORKLOAD MANAGER are connected by a bidirectional arrow. The SECURITY SUBSYSTEM is connected to both the DATA PARSER and the WORKLOAD MANAGER by bidirectional arrows. The SECURITY SUBSYSTEM receives RESULTS (132) as input and outputs WORKLOADS (130). A label 350 points to the overall system.

```
graph TD; DP[DATA PARSER 352] <--> WM[WORKLOAD MANAGER 358]; SS[SECURITY SUBSYSTEM 354] <--> DP; SS <--> WM; R[RESULTS 132] --> SS; SS --> W[WORKLOADS 130];
```

FIG. 3D

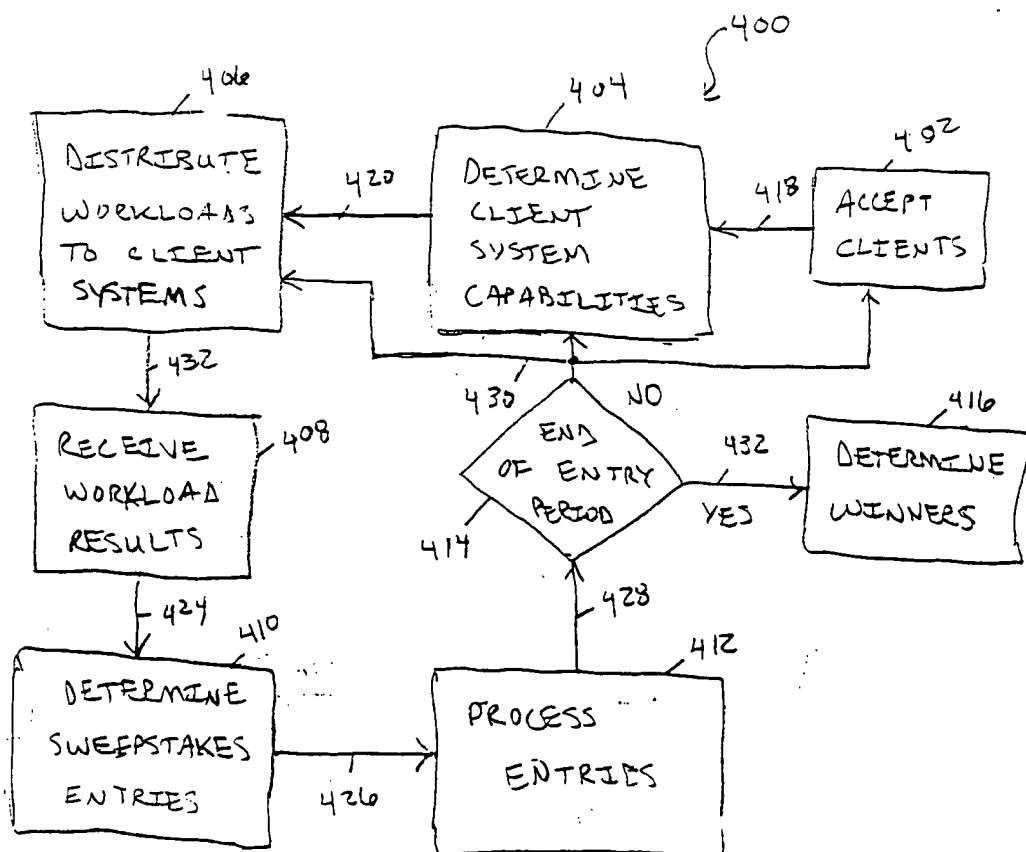
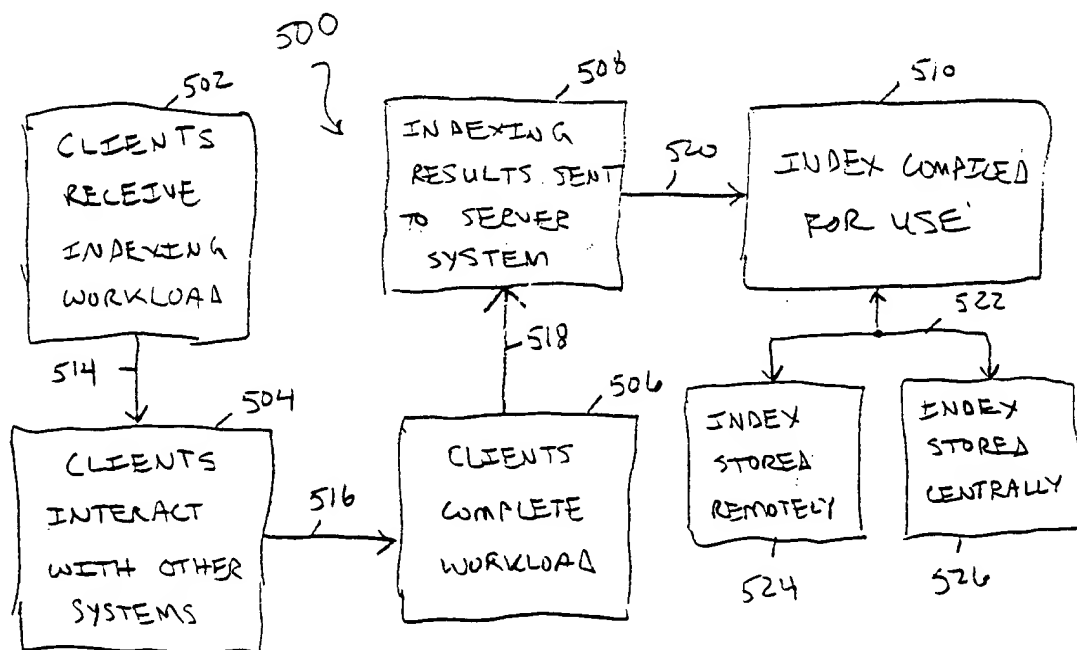
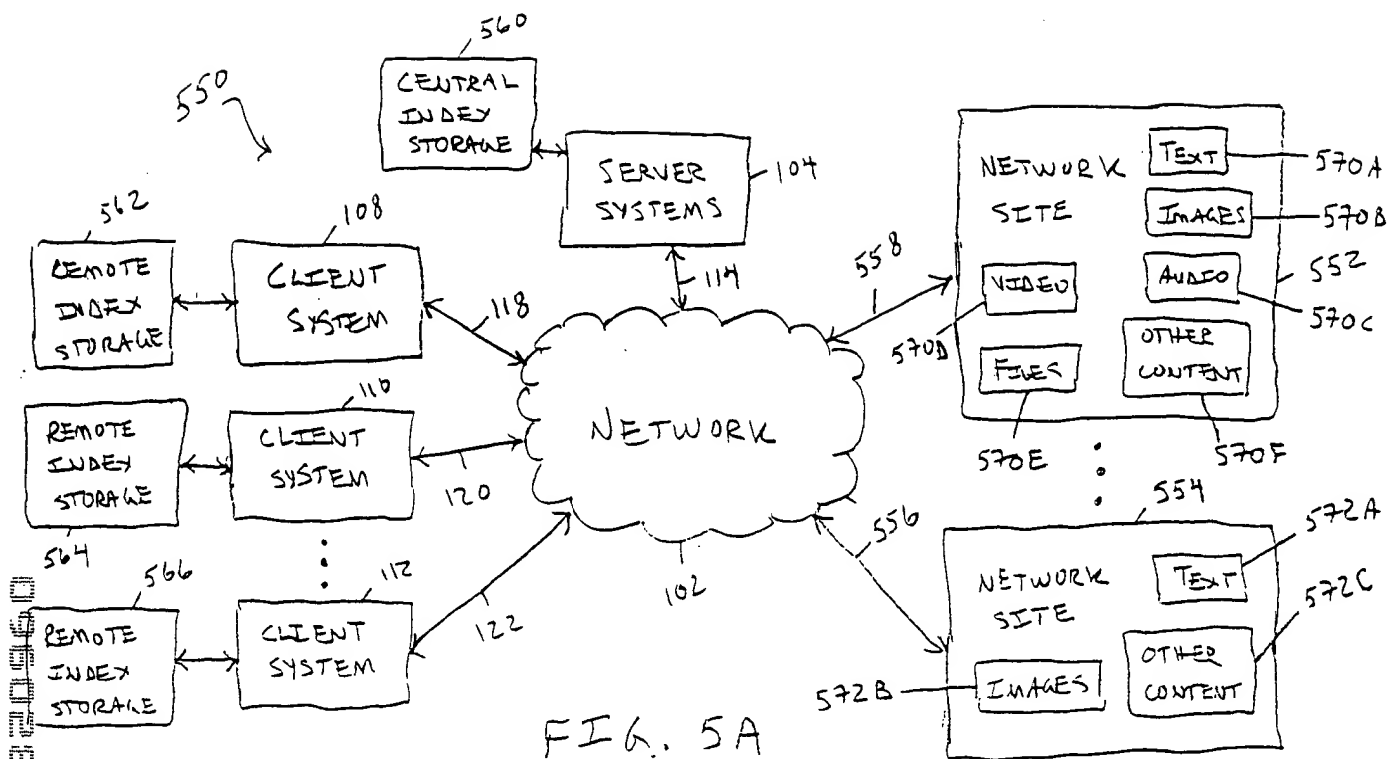


FIG. 4

~~SECRET~~ - REF ID: A66082



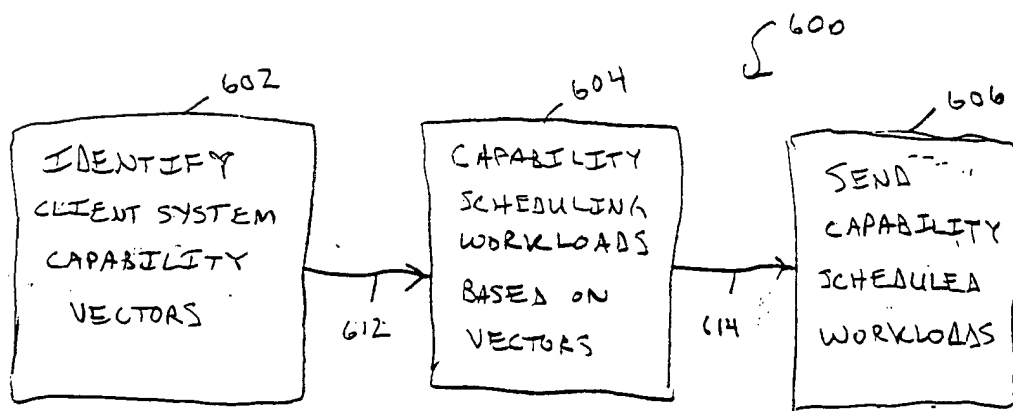
The diagram illustrates the system architecture with the following components and connections:

- 304 CONTROL SYSTEM**: The central control unit.
- 620 CAPABILITY VECTORS DATABASE**: A database containing capability vectors **CBV1**, **CBV2**, ..., **CBVN**.
  - Each vector box is labeled with a number: **628** for CBV1, **630** for CBV2, and **632** for CBVN.
- 308 WORKLOAD DATABASE**: A database containing workload entries arranged in a grid.
  - Row 1: **WL11** (labeled 640), **WL21** (labeled 646), ..., **WLN1** (labeled 652).
  - Row 2: **WL12** (labeled 642), **WL22** (labeled 648), ..., **WLN2**.
  - Row 3: Indicated by vertical dots.
  - Row 4: **WL1N** (labeled 644), **WL2N** (labeled 650), ..., **WLN** (labeled 654).

**Connections:**

- A bidirectional arrow labeled **626** connects the **CONTROL SYSTEM** and the **CAPABILITY VECTORS DATABASE**.
- An arrow labeled **624** points from the **CONTROL SYSTEM** to the **WORKLOAD DATABASE**.
- An arrow labeled **104** points from the **CAPABILITY VECTORS DATABASE** to the **WORKLOAD DATABASE**.

FIG. 6A



FTG, 6B

```
graph TD; A[CLIENTS RELIEVE TESTING WORKLOAD] -- 702 --> B[SITE TESTING RESULTS SENT TO SERVER SYSTEM]; B -- 710 --> C[SITE TESTING RESULTS COMPILED FOR USE]; C -- 706 --> D[CLIENTS COMPLETE TESTING WORKLOAD]; D -- 716 --> E[CLIENTS INTERACT WITH OTHER SYSTEMS]; E -- 714 --> A;
```

The diagram illustrates a cycle for handling site testing workloads. It begins with clients relieving testing workload, which leads to sending site testing results to a server system. These results are then compiled for use. From there, clients complete their testing workload, which involves interacting with other systems. This interaction feeds back into the initial state where clients relieve testing workload.

FIG. 8



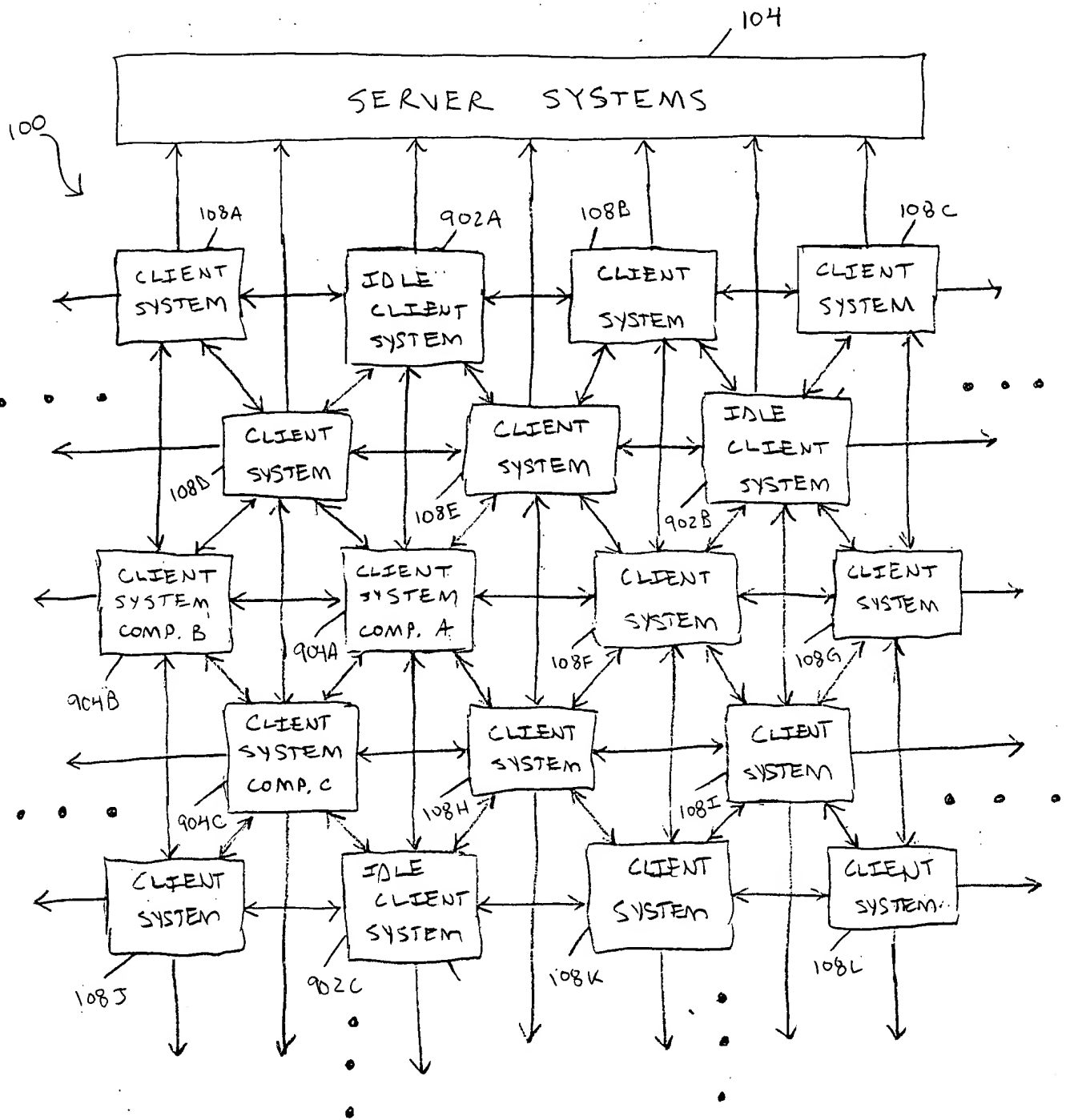
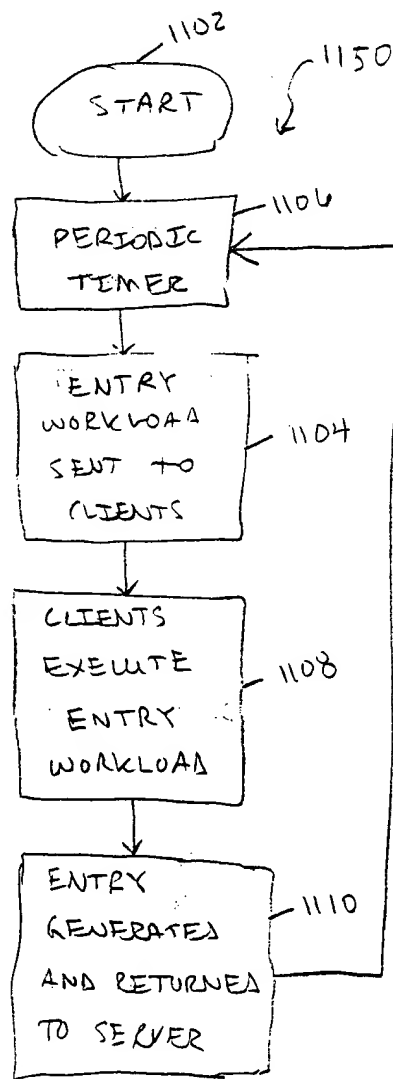
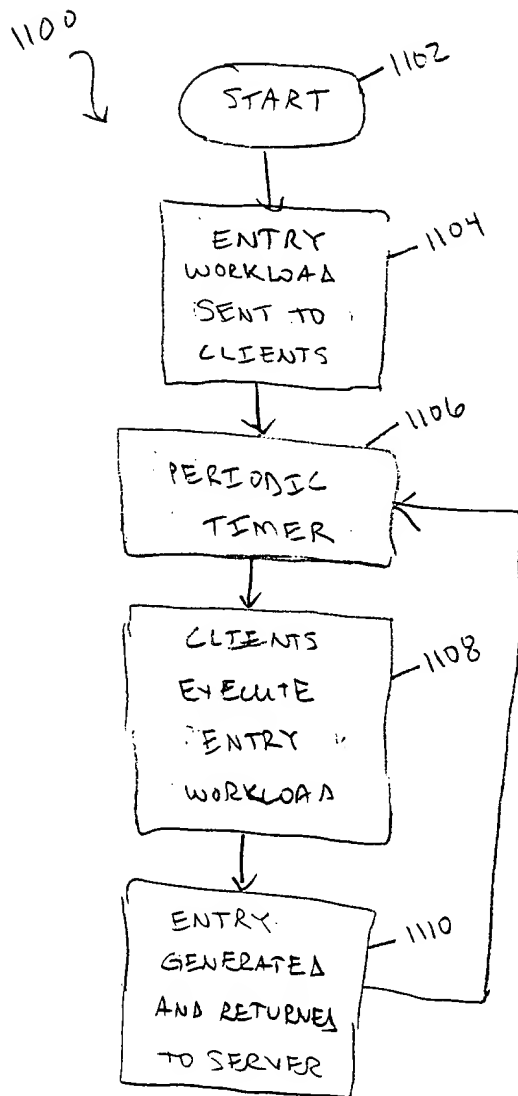
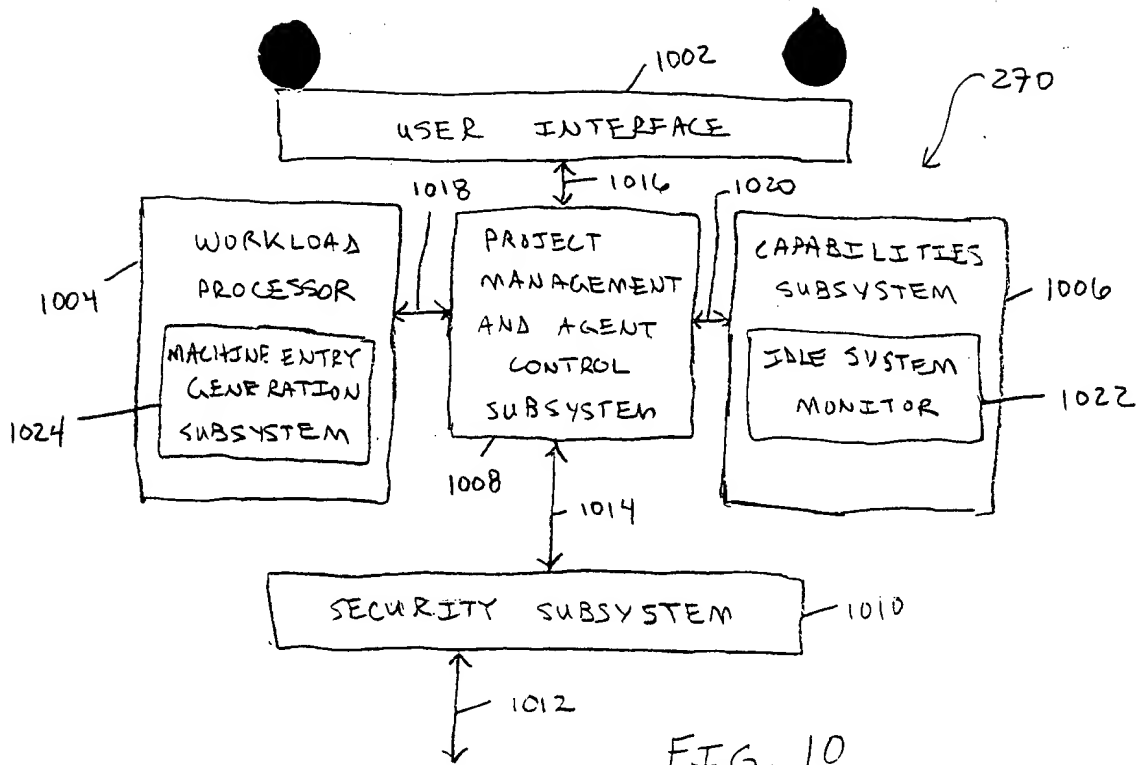


FIG. 9

00602303-06300

0960730 = 067000



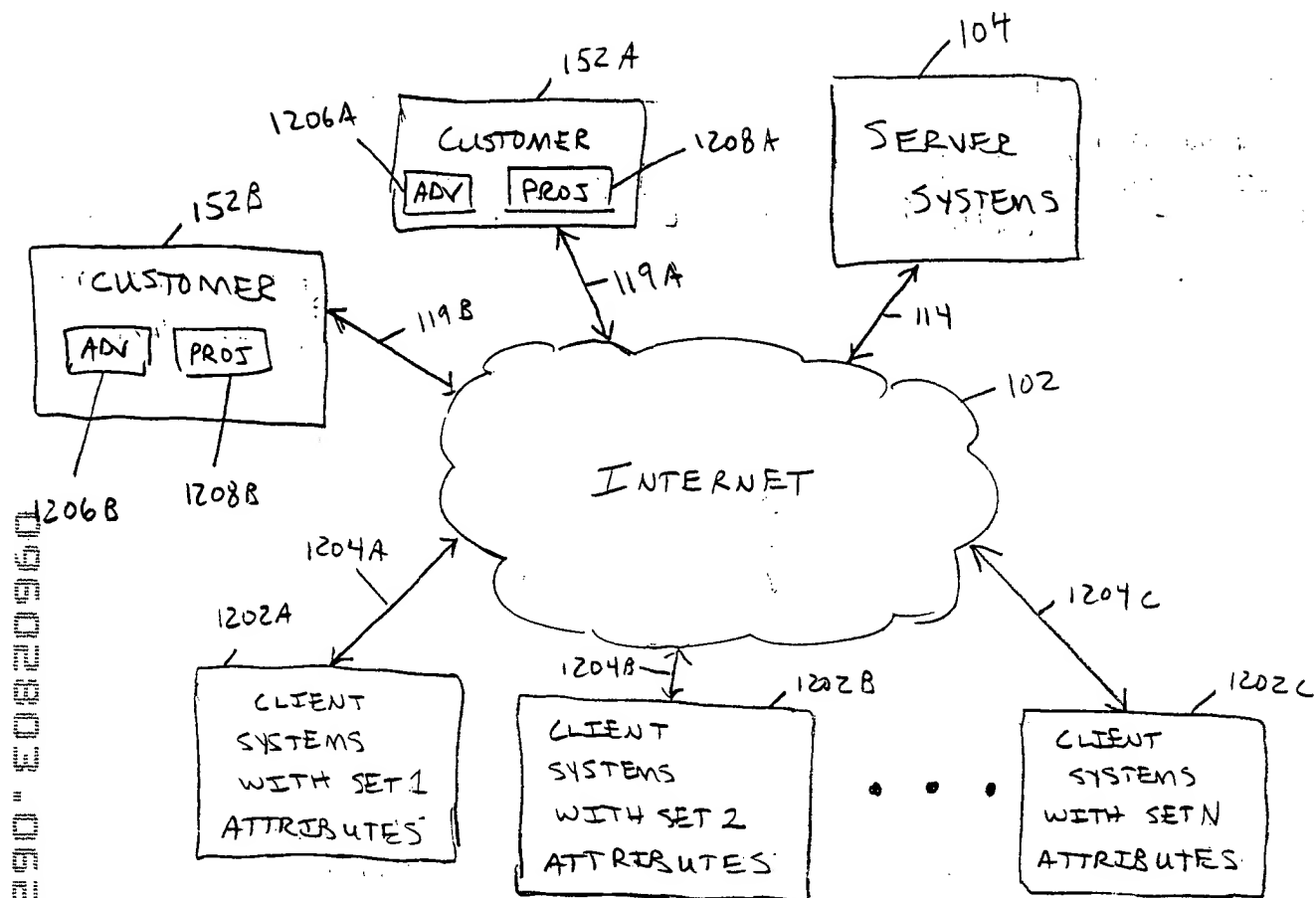


FIG. 12A

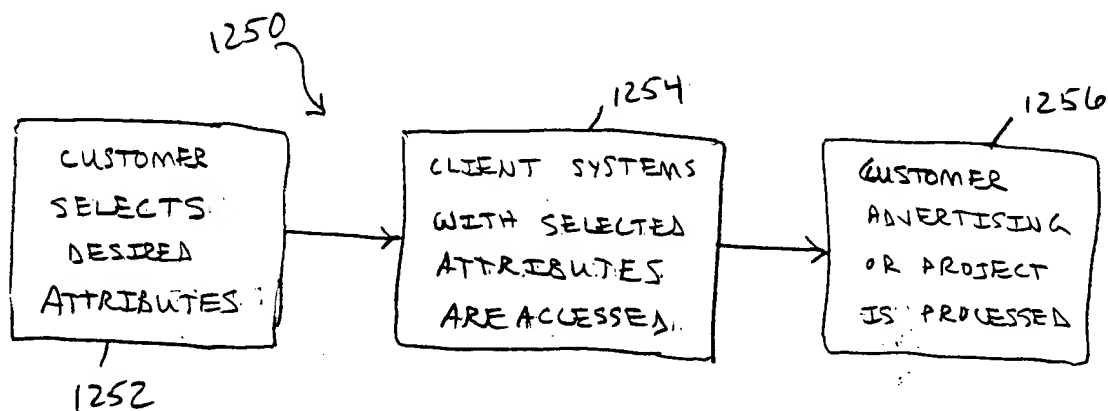


FIG. 12B



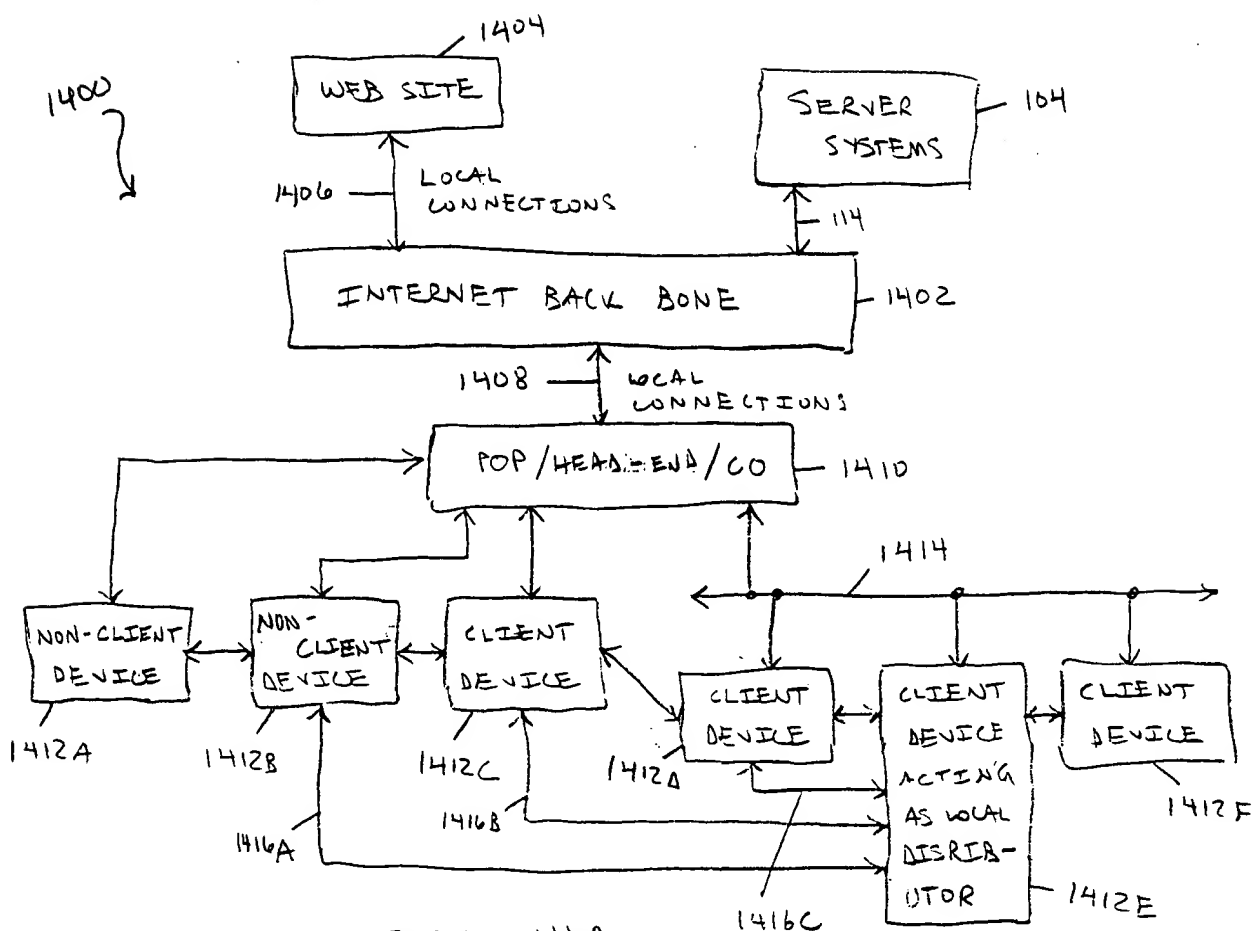


FIG. 14A

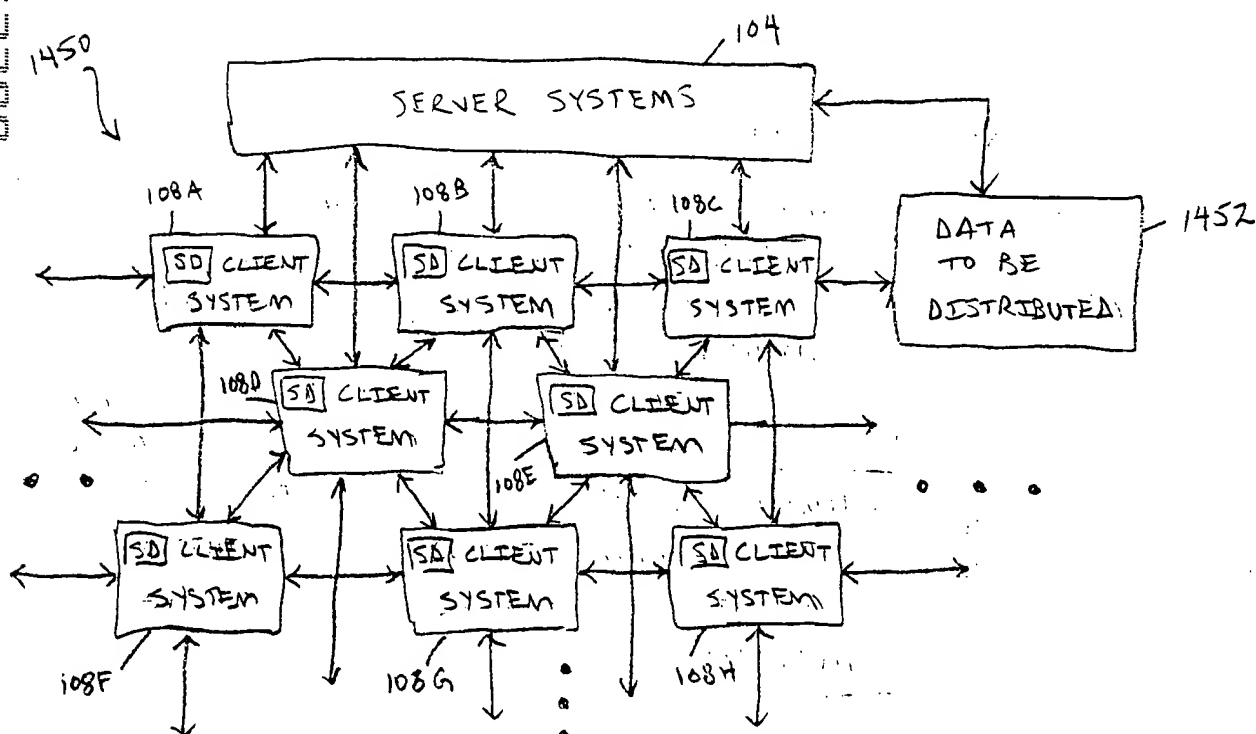


FIG. 14B

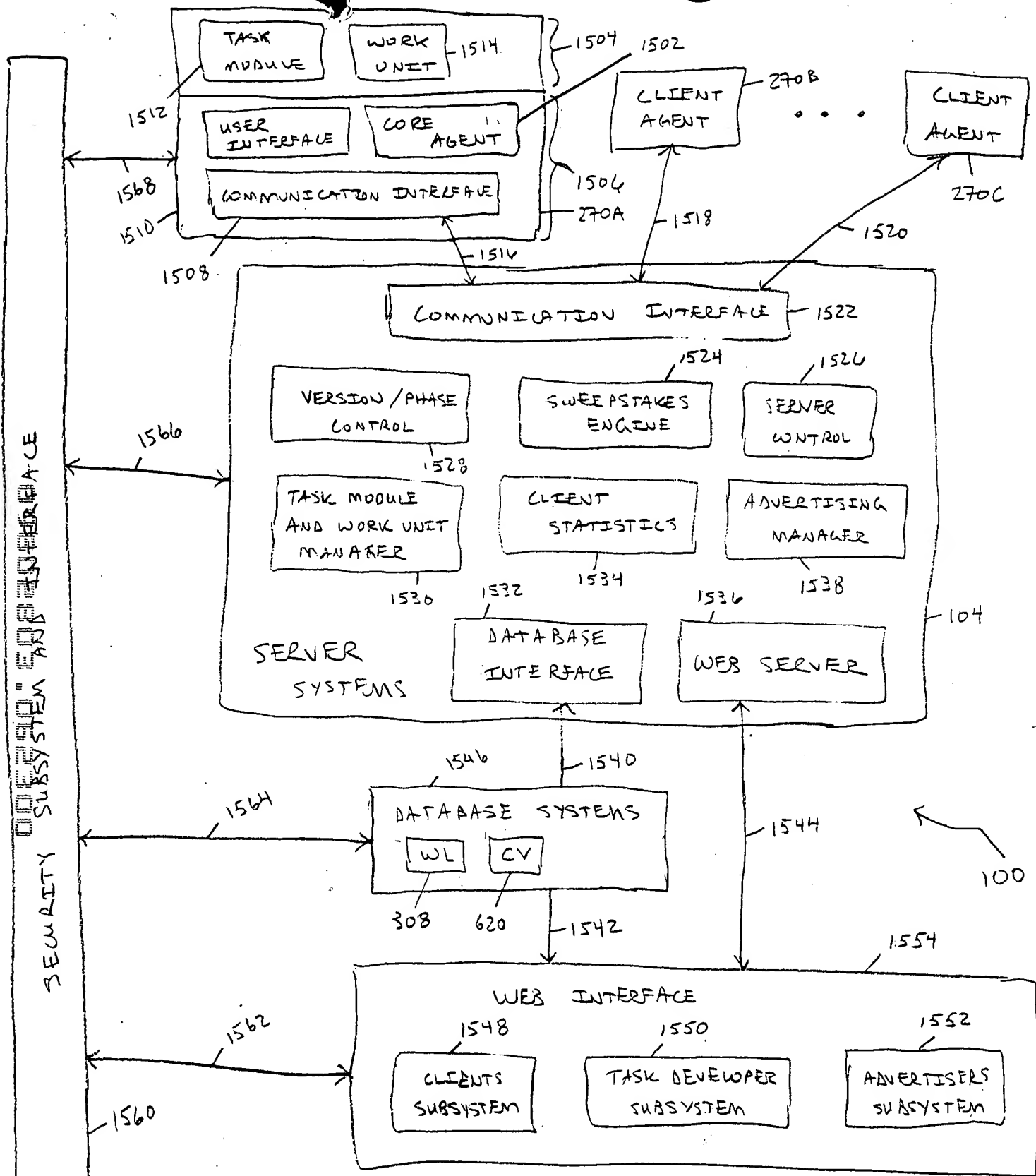


FIG. 15